REMARKS

By this amendment, Claims 27-28, 31-32, 38-41, 43-44, and 46-48 have been amended, Claim 49 has been cancelled, and no claims have been added. Hence, Claims 25-48 are currently pending in the application. Amendments to the claims are made to more clearly articulate the subject matter to which the Applicants desire patent protection. Support for the amendments to Claims 27-28 and 38-39 may be found, among other places, at page 11, lines 11-12. No new matter has been added.

SUMMARY OF THE REJECTIONS

Claims 25-49 are rejected under 35 U.S.C. § 103(a) as allegedly being anticipated by Ladd, Eric, et al., Using HTML 4, XML, and Java 1.2, 1999, Que. Platinum Edition (hereinafter "*Ladd*") in view of United States Patent Application No.: 2001/0011226 issued to Greer et al. ("*Greer*").

The rejections are respectfully traversed.

THE PENDING CLAIMS ARE PATENTABLE OVER THE CITED ART

Each of Claims 25-48 is patentable over the cited art because at least one element in each pending claim is not disclosed, taught, or suggested by the cited art.

Claim 25

Claim 25 recites:

> "storing a preconstructed web page;
> **storing, separate from said preconstructed web page, correlation data that specifies a correlation between an identifier and replacement content;**
> receiving a request for a requested web page that corresponds to said

preconstructed web page;
in response to said request, retrieving said preconstructed web page, wherein:
said preconstructed web page was created prior to receiving said request,
said preconstructed web page is written in a tag-delimited page description language, and
said preconstructed web page includes said identifier that is located at a position between a pair of tags within said preconstructed web page;
in response to said request, modifying said preconstructed web page to produce said requested web page by causing a program to perform the steps of:
removing said identifier from said preconstructed web page, and
inserting said replacement content at said position in said preconstructed web page, wherein said replacement content is selected based on the correlation data; and
providing said requested web page in response to said request." (emphasis added)

The above-cited combination of elements are not disclosed, taught, or suggested by *Ladd* or *Greer*, either individually or in combination.

The Office Action relies upon *Ladd* to show the above bolded elements. However, it is respectfully submitted that Active Server Page (ASP) do not operate as they are alleged to operate by the Office Action. As a result, *Ladd* cannot show what it is alleged to show by the Office Action.

How Active Server Pages (ASP) Work

Both *Ladd* and the pending claims are directed towards generating a web page, and both *Ladd* and the pending claims perform operations that modify a web page in response to receiving a request for a web page. However, beyond these broad generalities, there are sharp contrasts between the approach of *Ladd* and the approach recited in the pending claims.

The approach of *Ladd* is directed towards an Active Server Page (ASP) application that comprises scripted instructions embedded in an HTML document (see page 850 of *Ladd*).

In the approach of *Ladd*, when the server receives a request for the web page, the server may execute a script embedded in the web page. The script is called by an explicit call to the script. The script, when executed, may modify the content of the web page.

If a client requested the web page shown on page 851 of *Ladd*, then the server would execute the embedded VBScript entitled 'HelloWorld' when the web page was retrieved because the web page contains an explicit call to the VB Script "HelloWord." The explicit call is shown by:

"<%Call HelloWorld %> '===Calling the Active Server subroutine

As a result, the text "Hello World" would be generated because the VB Script 'HelloWorld' is designed to produce the text "Hello World" when the VB Script is executed. The text "Hello World" would appear in the requested web page, as served to the client, at the location where the VBScript 'HelloWorld' was called, namely at the location of "<%Call HelloWorld %>". However, the embedded VBScript entitled 'HelloWorld,' and the explicit call to the VB Script, would not appear in the requested web page as served to the client. This is so because the server parses out any VBScript before serving the requested web page to the client. *Ladd*, on page 851, states:

> when a browser requests a file containing an Active Server Page document, the server parses out and executes the scripted instructions. Any HTML output from the script is inserted back into the document in place of the original script code.

Unfortunately, the approach of *Ladd* suffers from exactly the same deficiencies that were described in the Applicants' background. Specifically, page 3, lines 5-16 of the Applicants' specification state:

> A second approach employs the reverse technique. Source code in a higher order programming or scripting language is placed into a Web page HTML code and is interpreted at runtime. Examples of this approach include

> ColdFusion, **ASP (Active Server Pages)** and JSP (Java Server Pages). Source code segments are placed directly into the HTML code. Tags or similar delimiters separate the source code segments from the page script. At runtime, a server executes the source code to generate a complete Web page. However, the source code is intrusive and can make using conventional Web editing tools difficult. The look and feel of a Web site can also be spread across several pages, imposing a maintenance burden. Finally, the application logic and user interface are not cleanly separated. Consequently, programmers and Web page artists find simultaneously working on the same page difficult. (emphasis added)

The approach of *Ladd* is directed towards the very same technique identified in the Applicants' background quoted above, namely Active Server Pages. Specifically, both the approach of *Ladd* and the portion of the Applicants' background quoted above feature the following same features:

1. Source code in a higher order programming or scripting language is placed into a Web page HTML code and is interpreted at runtime.
2. The scripting language or source code is an Active Server Page application.
3. Source code segments are placed directly into the HTML code.
4. Tags or similar delimiters separate the source code segments from the page script.
5. At runtime, a server executes the source code to generate a complete Web page.
6. The source code is intrusive and can make using conventional Web editing tools difficult.
7. The look and feel of a Web site can also be spread across several pages, imposing a maintenance burden.
8. The application logic and user interface are not cleanly separated.

Discussion of an illustrative embodiment

Advantageously, the approach taken by the pending claims solves the problems described by the Applicants' background and experienced by *Ladd.*

In the approach of the pending claims, **correlation data** that specifies a correlation between an identifier and replacement content **is stored separate from the preconstructed web page**.

The replacement content, identified by the correlation data, is inserted into the preconstructed web page at a position identified by an identifier. Because the correlation data is stored separate from the preconstructed web page, the problems associated with the approach of *Ladd* are avoided.

To illustrate how an embodiment operates, FIG. 3 of the Applicants' patent application illustrates a pre-constructed web page. At lines 11 and 12 of FIG. 3, the identifiers #HFEF#, #SYMBOL#, and #COMPANY# are recited. When the pre-constructed page is requested, a controller script 31 (shown in FIG. 2) removes the identifiers, and inserts replacement content in lieu of the corresponding identifier based on correlation data.

Correlation data is data that is stored, separate from the pre-constructed web page, which specifies a correlation between an identifier and replacement content. For example, controller script 31 may contain the correlation data, as shown below and on page 12, lines 5, 19 of the Applicants' specification:

> For example, a controller script 31 written in PL/SQL to generate a dynamic Web page from the HTT template 35 shown in FIGURE 3 is as follows:

```
BEGIN
htt.get ('companies.html');
htt.sub ('HREF', 'http://www.acme.com);
htt.sub ('SYMBOL, 'ACME');
htt.sub ('HREF', 'ACME');
htt.sub ('COMPANY, 'ACME Corporation');
htt.break;
htt.sub ('HREF', 'http://www.orcl.com);
htt.sub ('SYMBOL, 'ORCL);
htt.sub ('COMPANY, 'Oracle Corporation);
htt.showpage;
END;
```

As shown above, the correlation data in the controller script 31 associates the identifier HREF with replacement content (http://www.acme.com and http://www.orcl.com), associates the identifier SYMBOL with replacement content (ACME and ORCL), and associated the identifier COMPANY with replacement content (ACME Corporation and Oracle Corporation). As taught in the Applicants' specification at page 11, line 24 – page 12, line 4, the correlation data may specify that more than one piece of replacement content may have a correlation to a particular identifier, as shown above. The requested web page dynamically generated by the controller script 31 in this example is shown in FIG. 4.

The differences between the cited art and the claims

In view of the differences between *Ladd* and the approach of the pending claims, numerous elements featured in Claim 25 are not disclosed, taught, or suggested by *Ladd*. No portion of *Ladd* teaches, discloses, or suggests the element of "storing, **separate from said preconstructed web page**, correlation data that specifies a correlation between an identifier and replacement content" as featured in Claim 25. On the contrary, ***Ladd*** **expressly teaches away from this element** as anything analogous to both an identifier and replacement content in *Ladd* is embedded within the preconstructed web page. As a result, there would be no motivation in *Ladd* to maintain, separate from the preconstructed web page, any correlation data.

In the approach of *Ladd*, to the extent that anything is analogous to an identifier as claimed, the identifier must correspond to the explicit call to the embedded VB script that is embedded within the web page. This is so because the call to the VB script is removed from the web page, and therefore, satisfies the limitation of "removing said identifier from said preconstructed web page." Also, since the call to the VB script determines where any replacement content is to be inserted into the web page once the call to the VB script is served, the call to the VB script satisfies the limitation of "inserting said replacement content at said position."

Further, to the extent that anything is analogous to replacement content as claimed, the replacement content must be the result of processing the VB script embedded within the web page. This is so because the processing of the VB script embedded within the web page is the only mechanism capable of producing anything analogous to replacement content. For example, when the Hello World script of Listing 33.3 on page 851 of *Ladd* is executed, the only data specifying that "Hello World" is to be printed upon execution of the script is contained within the VB script entitled 'HelloWorld.'

However, in *Ladd*, no data is stored **separate from the preconstructed web page** that specifies a correlation between (a) the call to the embedded VB script, and (b) the result of processing the embedded VB script. Consequently, *Ladd* does not show correlation data as claimed, as correlation data is (a) stored separate from the preconstructed web page and (b) specifies a correlation between the identifier and the replacement content.

The Office Action attempts to show correlation data by explaining, "the code for ASP, which performs this function, is inherently stored separately from the web page." This is incorrect. There is no code in *Ladd* that is stored separately from the preconstructed web page that "specifies a correlation between an identifier and replacement content." In fact, as clearly shown on page 851 of *Ladd* and by the above discussion, the ASP code **must** be contained within the preconstructed web page. For example, 851 of *Ladd* indicates that "**all** script code is enclosed within the <% %> tags of the web page. While ASP scripts are executed by an entity separate from the preconstructed web page, the entity that executes the ASP script embedded within a preconstructed web page does not contain any data that specifies a correlation between an identifier and replacement content, and thus, cannot qualify as correlation data as claimed.

Consequently, *Ladd* cannot possibly disclose, teach, or suggest the feature of "storing, separate from said preconstructed web page, correlation data that specifies a correlation between an identifier and replacement content" as recited in Claim 25.

If the Office maintains the position that *Ladd* teaches the above element, the Office is respectfully requested to specifically identify what in *Ladd* is analogous to **each** of:

(a) an identifier as claimed,

(b) replacement content as claimed, and

(c) correlation data as claimed.

No portion of *Greer* is cited to show, nor does show, this element. Consequently, even if *Ladd* and *Greer* were to be properly combined, the resulting combination would still to disclose, teach, or suggest this feature.

Further, no portion of *Ladd* teaches, discloses, or suggests the element of "inserting said replacement content at said position in said preconstructed web page, wherein said replacement content is selected based on the correlation data" as featured in Claim 25. In sharp contrast, to the extent that content is inserted into a web page in the approach of *Ladd*, the content to be inserted is specified by data contained **within the embedded script** of the web page, (namely the VB script contained within the preconstructed web page) and not by correlation data **stored separate from the preconfigured web page**. Consequently, *Ladd* cannot possibly disclose, teach, or suggest this element.

No portion of *Greer* is cited to show, nor does show, this element. Consequently, even if *Ladd* and *Greer* were to be properly combined, the resulting combination would still to disclose, teach, or suggest this feature.

Consequently, as at least one element of Claim 25 is not shown, taught, or suggested by the cited art, it is respectfully submitted that Claim 25 is patentable over the cited art, and is in condition for allowance.

Claims 26-48

Claim 37 contain elements similar to those discussed above with respect to Claim 25, except that Claim 37 is recited in computer-readable medium format. Thus, it is respectfully submitted that Claim 37 is patentable over the cited art, and is in condition for allowance, for at least the reasons discussed above with respect to Claim 25.

Claims 26-36 and 38-48 are dependent claims, each of which depends (directly or indirectly) on one of the claims discussed above. Each of Claims 26-36 and 38-48 is therefore allowable for the reasons given above for the claim on which it depends. In addition, each of Claims 26-36 and 38-48 introduce one or more additional limitations that independently render it patentable.

Due to the fundamental differences already identified, to expedite the positive resolution of this case a separate discussion of the additional limitations is not included at this time, although the Applicants reserve the right to further point out the differences between the cited art and the novel features recited in the dependent claims.

CONCLUSION

The Applicants believe that all issues raised in the Office Action have been addressed and that allowance of the pending claims is appropriate. The Examiner is respectfully requested to contact the undersigned by telephone if it is believed that such contact would further the examination of the present application.

For the reasons set forth above, it is respectfully submitted that all of the pending claims are now in condition for allowance. Therefore, the issuance of a formal Notice of Allowance is believed next in order, and that action is most earnestly solicited.

To the extent necessary to make this reply timely filed, the Applicant petitions for an extension of time under 37 C.F.R. §1.136. If any applicable fee is missing or insufficient, throughout the pendency of this application, the Commissioner is hereby authorized to charge any applicable fees and to credit any overpayments to our Deposit Account No. 50-1302.

Respectfully submitted,
HICKMAN PALERMO TRUONG & BECKER LLP

Christopher J. Brokaw
Reg. No. 45,620
Date: May 24, 2006

2055 Gateway Place, Suite 225
San Jose, CA 95110
Telephone: (408) 414-1080, ext. 225
Facsimile: (408) 414-1076

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: **Mail Stop Amendment**, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

On May 24, 2006 By Angelica Maloney
Angelica Maloney